Initial value problems for pdes

We start with the diffusion equation in 1+1 dimension:

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}$$

We have already written down a discrete version of this.

Forward Time, Centred Space

$$\frac{u_j^{(n+1)} - u_j^{(n)}}{\Delta t} = \frac{D}{a^2}\left(u_{j-1}^{(n)} - 2u_j^{(n)} + u_{j+1}^{(n)}\right) \tag{1}$$

Stability analysis

von Neumann analysis (not rigorous)

Fourier transform in space: $u(x) = \sum_k e^{ikx} u(k)$
Each $u(k)$ evolves independently in time
(at least for linear problems with constant coeffs)
This gives the eigenmode evolution

$$u_k^{(n+1)} = \xi_k u_k^{(n)} \quad \Longrightarrow \quad u_j^{(n)} = u_0^{(0)} \xi_k^n e^{ikja} \tag{2}$$

To find amplification factor ξ_k, substitute (2) into finite difference equation

$$\xi_k \begin{cases} > 1 & \text{exponential growth, instability} \\ < 1 & \text{exponential damping, stability} \\ = 1 & \text{more detailed analysis needed} \end{cases}$$

von Neumann stability: $|\xi_k| \leq 1 \, \forall k$

Stability for FTCS

Inserting the eigenmode evolution (2) into the FTCS equation (1) gives

$$\frac{\xi_k^{n+1} - \xi_k^n}{\Delta t} = \frac{D}{a^2}\xi_k^n\left(e^{ik(j-1)a} - 2e^{ikja} + e^{ik(j+1)a}\right)$$

$$\xi_k = 1 + \frac{D\Delta t}{a^2}\left(e^{-ika} - 2 + e^{ika}\right) = 1 - \frac{4D\Delta t}{a^2}\sin^2\frac{ka}{2}$$

Since this is always ≤ 1 stability means that $\xi_k \geq -1$

$$\Longrightarrow \frac{4D\Delta t}{a^2}\sin^2\frac{ka}{2} \leq 2\,.$$

'Worst case': $\sin^2(ka/2) = 1$

Stability condition: $\dfrac{\Delta t}{a^2} \leq \dfrac{1}{2D}$

An example

The MatLab files `ftcs.m`, `gaussbc.m` and `ftcs_driver.m` solve the diffusion equation with the initial distribution

$$u_0(x) = u(x, t_0) = e^{-x^2/4Dt_0}\,, \quad -5 \le x \le 5\,, t_0 = 0.1\,,$$

and boundary conditions

$$u(\pm x_0, t) = \sqrt{\frac{t_0}{t}}\, e^{-x_0^2/4Dt}\,, \quad x_0 = 5\,.$$

The grid spacing in the x direction has been set to $a = 0.05$, and the diffusion constant $D = 1$.

`ftcs_driver(dt,t)` plots the solution for time step `dt` at time(s) `t`.

Run this with `dt=0.0012` and see what you get.
Then run with `dt=0.0013` and see what happens.

FTCS in 2+1 dimension

Our Ansatz is now

$$u_{jl}^{(n)} = u_0 \xi_k^n e^{ik_x j \Delta x} e^{ik_y l \Delta y} \tag{3}$$

For $\Delta x = \Delta y = a$ the FTCS scheme is

$$\frac{u_{jl}^{(n+1)} - u_{jl}^{(n)}}{\Delta t} = \frac{D}{a^2}\left(u_{j-1,l}^{(n)} + u_{j,l-1}^{(n)} + u_{j+1,l}^{(n)} + u_{j,l+1}^{(n)} - 4u_{jl}^{(n)}\right)$$

Inserting (3) gives

$$\begin{aligned}\xi_k &= 1 + \frac{D\Delta t}{a^2}\left(e^{-ik_x a} + e^{-ik_y a} + e^{ik_x a} + e^{ik_y a} - 4\right)\\ &= 1 - \frac{4D\Delta t}{a^2}\left(\sin^2\frac{k_x a}{2} + \sin^2\frac{k_y a}{2}\right)\end{aligned}$$

$$|\xi_k| \leq 1\,\forall k \Longrightarrow \frac{\Delta t}{a^2} \leq \frac{1}{4D}$$

A pain in the neck

We do not like the stability condition!

- We want to model features at large scales $\lambda \gg a$
  Typical diffusion time is $\tau \sim \lambda^2/D$
  $\longrightarrow$ need $n = \frac{\tau}{\Delta t} \sim \frac{\lambda^2}{a^2}$ time steps
- We want to improve accuracy by reducing a
  But if $a \longrightarrow a/2$ then $\Delta t \longrightarrow \Delta t/4$
  $\longrightarrow$ 8 times as much cpu time!

Can we improve on this?

Second order time derivative?

FTCS is first-order accurate in time, second order in space

What about using second-order differencing in time?

Centred Time Centred Space

$$\frac{u_j^{(n+1)} - u_j^{(n-1)}}{2\Delta t} = \frac{D}{a^2}\left(u_{j-1}^{(n)} - 2u_j^{(n)} + u_{j+1}^{(n)}\right)$$

von Neumann

$$\xi_k - \frac{1}{\xi_k} = -\frac{8D\Delta t}{a^2}\sin^2\frac{ka}{2}$$

$$\Longrightarrow \xi_k = -\frac{4D\Delta t}{a^2}\sin^2\frac{ka}{2} \pm \sqrt{1 + \left(\frac{4D\Delta t}{a^2}\sin^2\frac{ka}{2}\right)^2}$$

The $(-)$ mode is unstable for all k and Δt!

CTCS is unconditionally unstable

Implicit schemes: BTCS

Explicit scheme: $\frac{\partial^2 u}{\partial x^2}$ evaluated at t

Implicit scheme: evaluate at $t + \Delta t$

Backward Time, Centred Space

$$\frac{u_j^{(n+1)} - u_j^{(n)}}{\Delta t} = \frac{u_{j-1}^{(n+1)} - 2u_j^{(n+1)} + u_{j+1}^{(n+1)}}{a^2}$$

$$\implies \left(1 + \frac{2\Delta t}{a^2}\right) u_j^{(n+1)} - \frac{\Delta t}{a^2}\left(u_{j-1}^{(n+1)} + u_{j+1}^{(n+1)}\right) = u_j^{(n)}$$

We get a sparse matrix equation for $u^{(n+1)}$.

von Neumann analysis

$$\frac{\xi - 1}{\Delta t} = \frac{\xi}{a^2}\left(e^{ika} - 2 + e^{-ika}\right) \quad \implies \xi = \frac{1}{1 + \frac{4\Delta t}{a^2}\sin^2\frac{ka}{2}}$$

$\xi < 1$ for all $k, \Delta t$: BTCS is unconditionally stable

Crank–Nicolson

BTCS is stable, but only first-order accurate in time.
How can we get second-order accuracy?

Average FTCS and BTCS!
(the same as taking a centred time derivative around $t + \Delta t/2$)

$$\frac{u_j^{(n+1)} - u_j^{(n)}}{\Delta t} = \frac{1}{2a^2}\left[\underbrace{u_{j+1}^{(n+1)} - 2u_j^{(n+1)} + u_{j-1}^{(n+1)}}_{FTCS} + \underbrace{u_{j+1}^{(n)} - 2u_j^{(n)} + u_{j-1}^{(n)}}_{BTCS}\right]$$

Crank-Nicolson

$$\left(1+\frac{\Delta t}{a^2}\right)u_j^{(n+1)} - \frac{\Delta t}{2a^2}\left(u_{j-1}^{(n+1)} + u_{j+1}^{(n+1)}\right) = \left(1-\frac{\Delta t}{a^2}\right)u_j^{(n)} + \frac{\Delta t}{2a^2}\left(u_{j-1}^{(n)} + u_{j+1}^{(n)}\right)$$

Stability of Crank-Nicolson

von Neumann analysis

$$\xi\left(1 - \frac{-4\Delta t}{2a^2}\sin^2\frac{ka}{2}\right) = 1 + \frac{-4\Delta t}{2a^2}\sin^2\frac{ka}{2}$$

$$\implies \quad \boxed{\xi = \frac{1 - \frac{2\Delta t}{a^2}\sin^2\frac{ka}{2}}{1 + \frac{2\Delta t}{a^2}\sin^2\frac{ka}{2}}} \quad = \frac{1 - b^2}{1 + b^2}$$

The modulus of the numerator is always smaller than the denominator

Crank-Nicolson is unconditionally stable